

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

In the Specification

Please insert the following paragraph before line 1:

This application is a continuation of U. S. Patent  
Application Serial No. 09/613,118, filed July 10, 2000.

Please amend the paragraph beginning at page 2, line 16 as follows:

United States Letters Patent [Application Serial] No. [08/842,953] 6,101,497 of Ofek for a Method and Apparatus for Independent and Simultaneous Access to a Common Data Set, assigned to the same assignee as this invention, discloses a concept for making such an improvement. In accordance with that disclosure, certain physical disk drives in a disk array storage device are configured to be available to an application. These are called "standard devices". Other logical devices are configured to act either as a mirror for a standard logical device or to be split to provide a copy of the data on the standard device for some other purpose. In the context of the systems manufactured by the assignee of this invention, the second logical devices are called "BCV devices". Using the foregoing airline reservation systems as an example, the invention disclosed in United States Letters Patent [Application Serial] No. [08/842,953] 6,101,497 enables a BCV

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

device to attach to a standard device thereby to act as a mirror. Generally speaking, anytime after the BCV device has achieved synchronism with the standard device, the BCV device can be split, or detached, from the standard device. The copy of the data on the BCV device is then available to other applications, such as a backup application. This allows the other application to act on the data on the BCV device independently of and simultaneously with the continued operation of the main application with data stored on the standard device.

Please amend the paragraph beginning at page 3, line 15 as follows:

As the use of such data processing [systemS] systems has grown, certain issues that impact the splitting of a BCV device from its corresponding standard device have appeared. These include an issue of pending write data operations. Disk array storage devices of many manufacturers, including those of the assignee of this invention, utilize cache memory to enhance performance, particularly for write operations. When a host issues a write command, the data to be written transfers only to the cache memory before the operation is signaled to be complete back to the host. That data remains in the cache for some interval before that data, or overwritten data to the same

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

location, transfers to the logical device itself. During that transient interval in the cache, the operation is complete with respect to the host, but pending with respect to physical disk device. The entry in the cache is labelled as being a "write pending" entry. The process of transferring a "write pending" entry to a logical device is called "destaging".

Please amend the paragraph beginning at page 5, line 9 as follows:

United States Letters Patent [Application Serial] No. [09/303,242] 6,370,626 to Gagne et al. discloses a Method and Apparatus for Independent and Simultaneous Access to a Common Data Set that reduces this lock time by implementing an "instant split" operation. When an "instant split" command is received, the BCV device immediately detaches from the standard device and becomes accessible to an alternate application. This occurs under a lock condition that lasts in the order of microseconds during which certain control operations are accomplished but no data is transferred and no write pending entries are managed. Immediately thereafter the lock is released. Various processes in the disk array storage device thereafter manage the write pending entries in an orderly fashion even as the main application interacts with the

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

standard device and the alternate application, such as a backup application, interacts with the BCV device.

Please amend the paragraph beginning at page 7, line 18 as follows:

It is difficult at best [than] then to identify any such inconsistent data, particularly when dependent data transfers are involved. Consequently it becomes very difficult to recover data in the event of some type of malfunction. What is needed is a method and apparatus for enabling such instant split operations to occur such that related data on multiple split BCV devices or the like is consistent thereby to prevent any data corruption.

Please amend the paragraph beginning at page 10, line 21 as follows:

Hosts, such as the host 20 in FIG. 1, operate utilizing some commercially available operating system, such as the IBM MVS operating system. Such a host generally includes a plurality of control processors, with FIG. 1 depicting two control processors CP(1) and CP(n) identified by reference numerals 24 and 25, respectively. These control processors communicate with a main storage unit 26 that, as known, is divided into private, common, and extended private storage

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

areas. A control console device 27 permits an operator to communicate with the system for performing a number of configuration, diagnostic and other procedures independently of operations that occur in response to any application programs.

Please amend the paragraph beginning at page 12, line 1 as follows:

The bus 34 can be constituted by a single or multiple bus structure[,] and also connects to a cache memory 38. The cache memory 38 acts as a buffer and stores data including write pending data in accordance with a number of functions. Certain areas of the cache memory 38 will be devoted to cache slots for containing write pending entries and another area will be devoted to control.

Please amend the paragraph beginning at page 12, line 15 with the following:

Each disk adapter connects to one or more physical disk drives. Each physical disk drive may contain one or more logical volumes or devices. In addition a single logical volume may expand more than one physical device. For purposes of explanation, FIG. 1 depicts logical volumes particularly a logical volume 40 connected to the disk adapter [[34 ]]35, a

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

logical volume 41 connected to the disk adapter [[35 ]]36 and a  
logical volume 42 connected to the disk adapter [[36 ]]37.

Please amend the paragraph beginning at page 12, line 23 as  
follows:

It is assumed that the disk array storage device 21  
(DASD-1) has been configured by the system manager to include  
at least three BCV volumes 43, 44 and 45 as described in the  
above identified United States Letters Patent [Application  
Serial] No. [08/842,953] 6,101,497 and No. [09/303,242]  
6,370,626. It also is assumed that the configuration  
establishes the relationship such that the BCV devices 43, 44  
and 45 attach to the standard devices 40, 41 and 42  
respectively. Each disk array storage device in the data  
processing system, such as the DASD-n device 22, would have a  
similar or analogous structure and operation.

Please amend the paragraph beginning at page 13, line 8 with  
the following:

In accordance with this configuration, the TF application  
in the address space [[33 ]]32 can issue an ESTABLISH command  
or commands that enable a designated one or ones of the BCV  
devices 43, 44 and 45 to attach to their respective standard  
devices 40, 41 and 42 and then to come into synchronism. More

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

specifically, the TF application in address space 32 provides a mechanism for controlling BCV devices. Prior art versions of the TF application implement the previously described ESTABLISH, SPLIT, INSTANT SPLIT, MULTIPLE INSTANT SPLIT commands and other commands useful in controlling the operation of BCV devices. The TF application ~~[[is ]]~~in address space 32 is modified in accordance with the invention to enable a consistent split operation whereby a group of BCV devices with a common data set or application can be split with an assurance that the data on all the split BCV devices will be consistent. A specific implementation of a command included in such a BCV control, or TF application, will include an operator, a sequence number, a BCV address, a type of split operation plus optional arguments. In one form a command for implementing this invention is:

Please amend the paragraph beginning at page 14, line 18 as follows:

As previously indicated, there are several types of split operations available including the conventional split command as described in United States Letters Patent [Application Serial] No. [08/842,953] 6,101,497, a single instant split command as describe in United States Letters Patent [Serial] No. [09/303,242] 6,370,626 and a multiple instant split

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

command. Following the BCV address field, a field includes "CONS" to designate a consistent split operation of this invention. Other values at this field indicate other split operations.

Please amend the paragraph beginning at page 15, line 19 as follows:

FIG. 3 is an overview of the TF application in address space 32. Initially, step 50 establishes the buffers and data structures including a buffer 51 shown in FIG. 2 for request block (REQB) data structures that control ensuing operations. Step 52 calls a TFINIT initialization procedure. This procedure processes each TF application command, parses that command and produces one or more REQB data structures. If a command defines a range of BCV devices, the TFINIT initialization procedure will produce one REQB data structure for each BCV device. When the initialization procedure ends, the buffer 51 shown in FIG. 2 contains a plurality of REQB data structures. FIG. 2 depicts several such REQB data structures including an REQB(n) data structure [53] 54, an REQB(n[+]-1) data structure [54] 53 and an REQB(n+1)1 data structure 55 by way of example.

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

Please amend the paragraph beginning at page 17, line 12 as follows:

The PROCESS SPLIT REQUEST procedure called in step 65 performs three basic operations. First, it prevents any write operations to any of the standard device involved in the split. In an MVS operating system environment, the PROCESS SPLIT REQUEST procedure requests the MVS operating system to raise the IOS level for each standard device attached to a BCV device in the consistency group. Next the procedure issues the necessary INSTANT SPLIT commands to each of the BCV devices in the group as defined by the sequence number. Third, the system resets the IOS level for each standard device associated in the consistency group. This entire operation occurs in a short time interval and is transparent to an application program interacting with a standard device even when the split involves hundreds of BCV devices. As with the instant split operation of United States Letters Patent [Application Serial] No. [09,303,242] 6,370,626 the management of all the write pending entries occurs after the consistent split has occurred.

Please amend the paragraphs beginning at page 18, line 4 through page 20, line 19 with the following:

When the REQB data structures associated with all sequence numbers have been analyzed, step 61 transfers control to step

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

74. If any errors have occurred, step [[75 ]]75A generates an appropriate error message. Step [[75 ]]75A then passes control to step 76, as does step 74 if no errors exist; and step 76 then performs any housekeeping operations to end the instance of the TF application.

So long as input commands are available, step 83 will enable step 84 to read an input command, such as a split command identified above. Step 85 parses that command to obtain its operator and various arguments. Step 86 obtains the sequence number from the command and saves in the BCVSEQ# register 57. Next the TFINIT procedure 51 decodes the command. If the command is a split command, step 87 transfers control to step 90 that calls a process SPLIT REQUEST procedure. During this procedure as described in more detail later, other parameters of the split command are used to produce one or more REQB data structures representing the information in the split command. If the command is other than a split command, it is processed by a procedure 91 that is not disclosed in any detail because it forms no part of this invention. Once a command is processed, step 92 increments the BCVREQ# register 75 whereby the BCVREQ# register [[81 ]]75 maintains a running total of the number of commands that have been generated. Control then passes from step 92 back to step 83 to analyze another input command and this continues until all the commands have been

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

read. When all the commands have been read, step 83 shifts control to return the system to step 56 in FIG. 3.

When step 90 in FIG. 4 calls the PROCESS SPLIT COMMAND procedure, step 93 in FIG. 5 further decodes the parsed command generated [[a ]]at step 85 of FIG. 4 for the purpose of establishing various flags and values. Several flags that are important to this invention are identified. If the process is an instant split, as it will be for a consistent split operation, step 94 transfers control to step 95 that sets an INSTANT SPLIT flag 96 shown in FIG. 2. Step 97 transfers control to step 100 to set a CONSISTENT SPLIT flag 101. Normal operations of a storage controller include verifying the existence of a BCV device and its corresponding standard device particularly if the standard device is associated with multiple operating systems, such as multiple images of the MVS operating system. The bypass argument described above controls whether this test runs. If the optional bypass argument is set to an active state in the command, step 103 sets the BYPASS ON-LINE flag 104. Step 105 determines if there is a TIMEOUT to control a timeout flag 106 and a timeout interval register 107. If a timeout exists, step 105 diverts control to step 110 that sets the TIMEOUT flag 106. Step 111 stores the interval included in the command. Once the process in FIG. 5 establishes these flags, step 112 sets other flags corresponding to other

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

parameters that again form no part of this invention but are known to those skilled in the art.

Please amend the paragraph beginning at page 20, line 20 as follows:

As previously indicated, a single split command may identify one or more BCV devices. Step 113 determines whether there is a UCB (unit control block) for each BCV device in the group by testing for non-zero addresses. When this operation is complete, control transfers to step 114. Step 114 completes a REQB data structure for each BCV device identified in the split command being processed. That is, if the split command identifies a single BCV device, one REQB data structure will be completed. If three BCV devices are listed in sequence, step 114 will replicate the data for REQB data structure to provide a one REQB data structure for each of the identified BCV devices. Each REQB data structure will include the identification of its corresponding BCV device in a BCV DEVICE ID field [115] 116. When this process is complete, control passes back to step 92 in FIG. 4 to increment the BCVREG# register 81.

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

Please amend the paragraphs beginning at page 21, line 10 through page 22 , line 17 with the following:

In essence, the TFINIT initialization procedure 51 establishes a number of REQB data structures in the buffer 52 of FIG. 2 for each BCV device. Each REQB data structure identifies the type of operation, the BCV device that is being controlled and other relevant information. For purposes of this discussion, an INSTANT SPLIT flag 96, CONSISTENT SPLIT flag 101, BYPASS-ON flag 104 and TIMEOUT flag 106 and TIMEOUT [interval ]VALUE 107 are included in each such REQB data structure. As will now be apparent, this operation occurs in parallel with any interaction between the APPL-1 application 30 and the DASD-1 storage controller 21 particularly the standard devices 40, 41 and 42. Thus changes during the operation of the TF application in address space 33 attaches and splits BCV devices 43, 44 and 45 transparently.

When the TF application in address space 32 of FIG. 1 decodes a split command in step 64, step 65 calls a PROCESS SPLIT REQUEST procedure that obtains information about the various requests and issues an appropriate split command for that request. This procedure begins with step 120 in FIG. 6 that performs an initialization procedure and tests various parameters. In addition step 120 [[tests ]]indicates whether an operating system lock exists. It also sets a UCB address

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

and CCW (Channel Control Word) address for the request and sets various other request parameters. All these procedures are known in the art.

If the INSTANT SPLIT flag 96 for the REQB data structure being analyzed is set, step [[121 ]]122 transfers control to step [[122 ]]123 that calls an INSTANT SPLIT procedure. The INSTANT SPLIT procedure determines whether there are other REQB data structures in the buffer 51 that have the same sequence number as established by the sequence number entry 123 in FIG. 2. This assures that multiple INSTANT SPLITS at the same level will be processed in one system calling procedure.

Please amend the paragraphs beginning at page 23, line 9 through page 24, line 7 with the following:

If the REQB data structure defines other than a CONSISTENT SPLIT operation, control transfers from step 130 to step [[134 ]]133. Control also passes from step 133 to step 134 to set an INSTANT DONE flag 135. Step 136 increments the value in the NUM PAIRS register 127 thereby to indicate that one standard-BCV device pair exists.

Now the INSTANT SPLIT procedure looks at all the remaining REQB data structures to determine if any other REQB data structures in the buffer [[52 ]]51 should be grouped with the REQB data structure being analyzed for a consistent split.

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

Step 137 begins a loop that examines these next REQB data structures. Initially step 140 transfers control to step 141 that determines if the sequence number for this next REQB data structure has the same value as the sequence number for the REQB data structure being analyzed. If it does, step 142 determines whether the next REQB data structure has a INSTANT SPLIT component. If it does, step 143 determines whether any errors exists. If none exist, step 144 looks to the INSTANT DONE flag in the next REQB data structure to determine whether it is set. If it has been set, there is no need to further process that particular REQB data structure. Next step 145 determines whether the CONSISTENT SPLIT flag for this next REQB data structure is set. Step 146 then determines whether the same disk array storage device is involved.

Please amend the paragraph beginning at page 25, line 16 as follows:

Assuming the [in-split] INSTANT SPLIT procedure completes successfully, control passes from step 153 to step 155 that again tests the CONSISTENT SPLIT flag 101. If the CONSISTENT SPLIT flag is set, and step 156 determines that the request is not pending from activities involving another storage controller, such as the DASD-n storage controller in FIG. 1, a test is made to determine whether the IOS level, an MVS

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

operating system feature, is raised in step 157 by testing an IOS LEVEL flag 158 in FIG. 2. If it has previously been raised, no action occurs. Otherwise step 160 calls a SET IOS LEVEL procedure. If the procedure completes in a positive fashion, step 161 transfers control to step 162. If for any reason the REQB data structure either is not for a CONSISTENT SPLIT, is not pending or already has its IOS LEVEL flag raised, control transfers to step 162 without setting the IOS LEVEL flag.

Please amend the paragraph beginning at page 28, line 10 as follows:

If no errors occur, step 177 passes control [passes] from step 176 to step 180 that issues a query to obtain the status of all the BCV devices [in] and step 181 that checks the state of each BCV device. Steps 180 and 181 essentially determine whether any BCV devices in any consistency group remains in a transition state that exists from the time the application issues a STARTIO command until an addressed BCV acknowledges the receipt of a command. Steps 180 and 181 operate in an iterative fashion until all operations are complete or until the STIMER function times out.

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

Please amend the paragraph beginning at page 29, line 1 with the following:

Step 185 in FIG. 9 determines whether the STIMER timed out. Specifically if the operation for the RUN STARTIO procedure does not terminate within the allotted time, a timeout occurs. When STIMER timeout occurs in an MVS operating system, another MVS operating system utility detects that event and initiates appropriate responses. Within the TF application, however, further responses are needed whether the STIMER times out or not. If it does not, control passes from step 185 to step 186 to reset the timer. In any event step 187 then determines the number of requests for which the IOS level has been set. Step 190 identifies a first request that is selected in step 191. Steps 192, 193, 194 and 195 then test various features to determine whether to issue an IOS LEVEL RESET command in step 196. Specifically if the CONSISTENT SPLIT flag 101 is set, an IOS LEVEL flag 158 is set, the sequence number matches the sequence number set forth in step [[133 ]]86 in FIG. 4, and the UCB address is not zero, step 196 issues an IOS LEVEL RESET command and clears the corresponding IOS LEVEL flag otherwise the IOS level flag is not altered. If this does not occur satisfactorily, step 197 transfers control to step [[200 ]]206 that prepares an appropriate error message.

Applicant(s): DOUGLAS E. LECRONE AND EUGENE D. PFLUEGER  
Serial No.: Continuation of 09/613,118 E30-046CON (00-87CON)  
Filed: Herewith

Thereafter step 201 determines whether more REQB data structures need review.

Please amend the paragraph beginning at page 31, line 1 as follows:

The advantage gained is that the data is consistent. Consequently if a problem occurs in the operation of the standard devices, recovery of the data backed up from the BCV devices after [[they ]]a consistent split, is greatly facilitated because it can be assumed that the data is coherent. This assures that any errors are readily detectable. For example, with a dependent data transfer, even if a split occurs after the first log entry or after the data is transferred but before the second log entry is made, a diagnostic program that monitors the log can easily determine that the second log entry is not present and then either automatically reconstruct the data associated with the log or complete the log. Other enhancements to the recovery and other processes will also be apparent to those of ordinary skill in the art as a result of having this consistent data back up on multiple BCV devices.